O-MI/O-DF vs. MQTT: a performance analysis

Paul-Lou Benedick, Jeremy Robert, Yves Le Traon
University of Luxembourg
Interdisciplinary Centre For Security, Reliability and Trust
Luxembourg
Email: {*firstname.lastname*}@uni.lu

Sylvain Kubler
*Université de Lorraine, CRAN, UMR 7039, Campus Sciences,
BP 70239, Vandoeuvre-lès-Nancy F-54506, France
+CNRS, CRAN, UMR 7039, France
Email: s.kubler@univ-lorraine.fr

***Abstract*—Over the past decade, a flourishing number of concepts and architectural shifts appeared such as Industrial Internet of Things (IIoT), Industrial CPS or even Industry 4.0. Unfortunately, today's IoT as well as Industry 4.0 environments, look more like a collection of isolated "Intranets of Things", also referred to as "vertical silos", rather than a federated infrastructure. Breaking down these silos is a key challenge in both the IoT and Industry 4.0 communities. This paper is intended to present and discuss two open and standardised messaging protocols designed for IoT applications, namely: MQTT and O-MI/O-DF. First, a traffic load's analytical model derived from the MQTT standard specifications is presented. Second, a comparison study between MQTT and O-MI/O-DF standards is carried out based on a real-life industrial implementation. This study brings a deep understanding of the extent to which these protocols are performant (from a traffic load perspective) and how they can impact on future architectural designs.**

I. INTRODUCTION

Over the past decade, a flourishing number of concepts and architectural shifts appeared such as Internet of Things (IoT), Cyber-Physical Systems (CPS) or Internet of Everything (IoE). Applying these concepts to the industrial application scenarios leads to the definition of the following terms: Industrial Internet of Things (IIoT), Industrial CPS or even Industry 4.0. Currently, the Industrial Internet consortium is essentially driven by US enterprises, meanwhile in Europe similar initiatives have different names: 'Industrie 4.0' in Germany, 'Smart Factory' in the Netherlands, 'Usine du Futur' in France, etc. Those disciplines have become a technological focus area for academia, industry, and governmental organisations, as stated by the number of papers found on Google Scholar: 287 papers – from 2013 – using 'Industrial Internet of Things' as title keywords, and 1160 papers using "Industry 4.0". Strictly speaking, differences between the aforementioned terms could be elicited. Nevertheless, the term Industry 4.0 is used throughout this document for consistency purposes.

One of the important aspects of the Industry 4.0 is to increase connectivity between the technologies, systems and processes [1]. Indeed, most of the companies that have been created decades ago, have heterogenous, non-interoperable and proprietary systems that are expensive to train on and maintain. Adding to that, future production systems have to be developed considering the need for strong product individualisation/customisation and, therefore, the necessity for high flexible production processed [2]. It includes the industrial communications networking and IT infrastructures. Those infrastructures are still based on the building automation pyramid [3], which are still too rigid for meeting flexibility and adaptability requirements. From a long-term perspective, networked Things and IO modules (at the field level) will be identifiable and accessible through the Internet, constituting a dynamic global network infrastructure with self-configuring capabilities. Even if a first level of interoperability is achieved at the field level by relying on open technology standards such as Ethernet-based solutions, it still remains to manage one of the most critical obstacles, namely the *vertical silos'* model that shapes today's IoT [4]. Data is not anymore dedicated to a particular use but expected to be connected to many other organizational information systems to support various types of activities, spanning from production, to business, and to services. One solution to solve this problem is to rely, as much as possible, on open communication standards at the Application layer, where both technical and semantic interoperability are tackled [5].

While many manufacturing companies are willing to move towards the Industry 4.0 paradigm, their IT infrastructure, often dating back from the early 70's or 80's, prevent them from taking full advantage of that paradigm. As a result, it is of the utmost importance to help them to efficiently step into the Industry 4.0 by proposing efficient network infrastructure frameworks to connect all the existing vertically-oriented closed systems (production units, metrology station, . . .). To do so, it is important for system designers to be aware, beforehand, about the traffic load that a system (through its gateway) will generate depending on the adopted communication protocol. The contribution of this paper is twofold: i) propose an analytical model of the traffic load (and efficiency ratio) based on the MQTT standard specifications ii) carry out a comparison analysis with an other well-know IoT communication protocol, and particularly the O-MI/O-DF standards (a first analytical model being proposed in [6]). These two protocols have been selected for this study not only because they can be used for Machine-to-Machine communications, but also due to their intrinsic capabilities. The first one implements an aggregation-like mechanism, meaning that all the data are bundled in the same message, whereas the second is not fully compliant with this model (using one response message per data item/topic). This study will allow us to draw conclusions about the efficiency of each protocol and their underlying mechanisms. The originality of this paper, compared with the

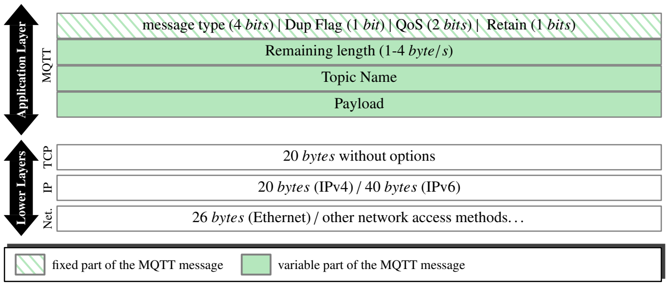


Fig. 1. MQTT message size

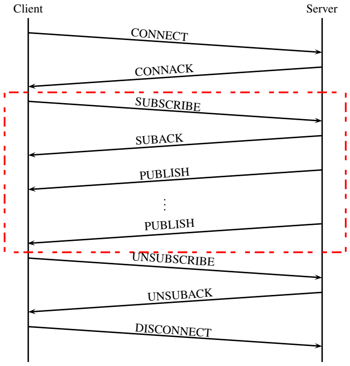


Fig. 2. MQTT sequence diagramm (considering a QoS level=0)

existing literature where several experimental analysis have been conducted with regard to MQTT [7], [8], lies in the fact that no comparison study between MQTT and O-MI/O-DF has been proposed and quantified yet.

The rest of the paper is organized as follows. Section II presents the mathematical models of the traffic load for both protocols. In Section III, a real industrial case-study is described and analysed. Section IV discusses on how to use these study results in an industrial CPS. Finally, Section V concludes this paper.

II. O-MI/O-DF &. MQTT: A TRAFFIC LOAD AND EFFICIENCY RATIO MODELS

A. *Introduction of O-MI/O-DF & MQTT*

O-MI [9] and O-DF [10] standards, which have been specified and published by *The Open-Group* standardization fora, are independent entities that reside in the OSI Application layer, respectively specified at the 'communication' and 'format' levels [11]. O-MI provides a generic Open API (specifying different interfaces such as Read, Write... as summarized in Fig. 3) for any RESTful IoT information system. IoT gateways that implement O-MI can act both as a "server" and "client", meaning that communications are established in a peer-to-peer manner. O-DF standard can be combined on top of O-MI - *although not mandatory* - for describing 'Things' in a generic manner. Note that more specific vocabularies can also be added to the O-DF structure, as discussed in [12].

MQTT is a connectivity protocol for IoT and Machine-to-Machine communications. Standardised by the OASIS body, it also resides in the OSI Application layer, relying on the Publish/Subscribe model (i.e., clients communicate with a broker in a peer-to-peer manner). The data are described according to a string-based (hierarchical) topic (e.g., "smartHouse/temperature"). MQTT distinguishes between three QoS levels at the application layer, since it relies on TCP/IP for the lower ones. In addition and contrary to O-MI, MQTT is a connection-oriented protocol, meaning that clients need to setup a connection with the broker before publishing or subscribing any data/topic.

B. *MQTT: a traffic load analytical model*

MQTT is independent of the lower layers (Medium access, Link layer, Network and Transport). Thus, the size of the request, or response, or acknowledgement, respectively denoted by S_{req}, S_{resp}, S_{ack} can be formalized as in Eq. 1.

$$S_{\text{req}} = \ell_{\text{low-layer}} + \ell_{\text{app-layer}} \tag{1}$$

1) Application layer: According to the OASIS standard specifications [13], the MQTT header is composed of both a fixed and variable part as amphasized in Fig. 1. The fixed part corresponds to the first byte and enables to specify the message type and QoS level. The variable part defines data-related information such as the header's length, topic's name, or data payload. The application layer size can therefore be written as in Eq. 2, where the remaining length is defined as the sum of the data and topic sizes. This value (in decimal) needs to be coded over 1 to 4 *bytes* according to the standard. As a consequence, the number of bytes needed for coding this decimal value can be expressed as in equation 3). Let us note that only 7 bits out of 8 are used to code the remaining length, as the first bit (called "*Continuous bit*") enables to specify whether or not there is a subsequent byte for this field.

$$\ell_{\text{app-layer}} = 1 + \ell_{\text{length}} + \ell_{\text{topic}} + \ell_{\text{data}} \tag{2}$$

$$\ell_{length} = \left\lceil \frac{\left\lceil \frac{\ln(\ell_{topic}+\ell_{data})}{\ln(2)} \right\rceil}{7} \right\rceil \tag{3}$$

2) Lower layers: As previously mentioned, MQTT takes place over TCP/IP connections. The length of the (network) IP header depends on whether IPv4 or IPv6 is in use, as emphasized in Fig. 1. In our model, we consider Ethernet as the underlying network access protocol, but other protocols coud be considered as well (e.g., IEEE 802.15.4). As a result, $l_{low-layer}$ is either equal to 66 bytes $(26 + 20 + 20)$ or 86 bytes $(26 + 40 + 20)$.

3) Traffic Load & Efficiency Ratio: As outlined in red in the Fig. 2, only the data exchanges phase (i.e. *Publish/Subscribe* messages) are considered in this study, which corresponds to a QoS level equal to zero. This is the best case for minimising the number of exchanges, and accordingly the traffic load. Connection and disconnection phases are not taken into account, but could easily be added since MQTT messages use the same header. Given this, let T be the number of topics that can be subscribed to (i.e., number of 'subscription' messages);

TABLE I
VARIABLES USED FOR O-MI/O-DF FORMULAS

Protocol	Variable	description
HTTP	ℓ_{url}	URL length
	ℓ_{reason}	HTTP reason-phrase length
O-MI	ℓ_{ttl}	O-MI TTL field length
	ℓ_{rc}	O-MI Return code length
	ℓ_{reqID}	Request ID length
	ℓ_{int}	Subscription Interval length
	ℓ_{call}	Callback address length
O-DF	ℓ_{objID}	Number of digits of Object's ID
	ℓ_{name}	Number of digits of InfoItems' name
	ℓ_{value}	Number of digits of Value

'Suback' be the message used by the broker to acknowledge each subscription; and P be the number of 'Publish' messages sent by the MQTT broker to all clients having subscribed to a given topic. It can be noted that $T \leq P$ since MQTT can use an aggregation mechanism in the subscription process. It means that the subscriber can directly subscribe to all the data hierarchy available at the broker level using the # character (e.g., `/smartHouse/#/` means that the request subscribes to all topics under the `smartHouse` topic) or using the wildcard + (e.g., `/smartHouse/+/temperature/` means that + will be replaced by any available Object). However, it cannot be used in the *Publish* messages, and it is therefore necessary to send as many messages as topics. Given all these parameters, the traffic load can be expressed as in the equation 4.

$$TL(T,P) = \sum_{t=1}^{T} \left(S_{\text{req}}(t) + S_{\text{suback}}(t)\right) + \sum_{p=1}^{P} S_{\text{resp}}(p) \tag{4}$$

Let us note that equation 4 (or Eq. 1 to be exact) does not take into account the lower layer constraints, and particularly the network access method in terms of Maximum Transmission Unit (MTU - 1500 $bytes$ in Ethernet). Indeed, if $\ell_{\text{app-layer}} > MSS$ (Maximum Segment Size - 1460 bytes with IPv4/TCP), the number of frames is expressed as $n = \left\lceil \frac{\ell_{\text{app-layer}}}{MSS} \right\rceil$. Eq. 1 can therefore be refined as in Eq. 5 to express the total length of data transmitted by the network (either for the request, response or the application acknoledgement).

$$L_{\text{req}} = (n-1) \cdot (\text{MTU} + \ell_{net}) + S_{\text{req}} - (n-1) \cdot \text{MSS} \tag{5}$$

As MQTT relies on TCP as transport protocol, it is also important to take into account the exchanges added by TCP such as the opening and closing TCP connections and segment acknowledgments. However, the transient states of TCP opening and closing operations are not considered in this study as one or more MQTT messages can be transmitted over a same TCP connection. The overall traffic load can therefore be defined as in Eq. 6, with L_{ack} defined as in Eq. 7

$$TL(T,P) = L_{\text{ack}}(T,P) + \sum_{t=1}^{T} L_{\text{req}}(t) + L_{\text{suback}}(t) + \sum_{p=1}^{P} L_{\text{resp}}(P) \tag{6}$$

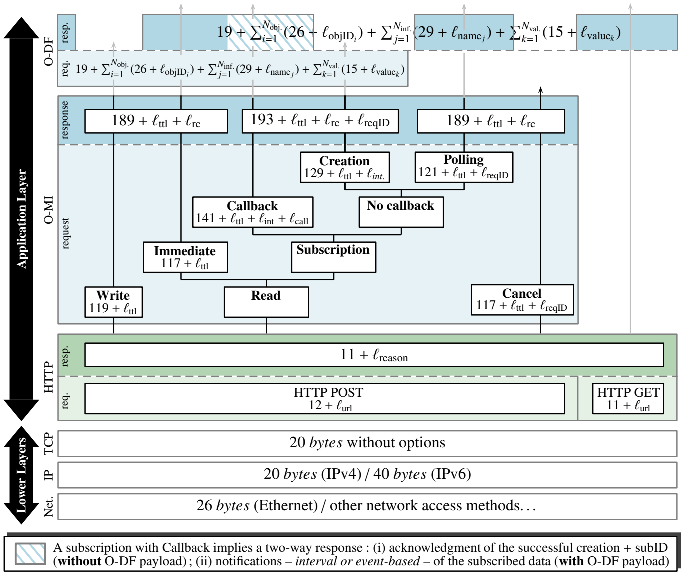


Fig. 3. O-MI/O-DF request/response message size [6]

$$L_{\text{ack}}(T,P) = \left[\sum_{t=1}^{T}\left(\left\lceil\frac{n_{\text{req}}(t)}{m}\right\rceil + \left\lceil\frac{n_{\text{suback}}(t)}{m}\right\rceil\right) + \sum_{p=1}^{P}\left(\left\lceil\frac{n_{\text{resp}}(p)}{m}\right\rceil\right)\right] \cdot \ell_{\text{low-layer}} \tag{7}$$

Eq. 7 takes into account the TCP acknowledgments, which can be sent either immediately a segment is received, or after several segments are received, or inside a new data transmission (piggybacking). This is taken into consideration in our study by defining the m variable as the number of received segments after which an acknowledgement is sent. Let us note that L_{ack} considers the acknowledgements for all messages used in the data exchanges.

Let us note that based on the traffic load, it is straightforward to define the efficiency ratio of the protocol. As this parameter is computed in the industrial case-study, Eq. 8 provides the generic expression of the efficiency ratio.

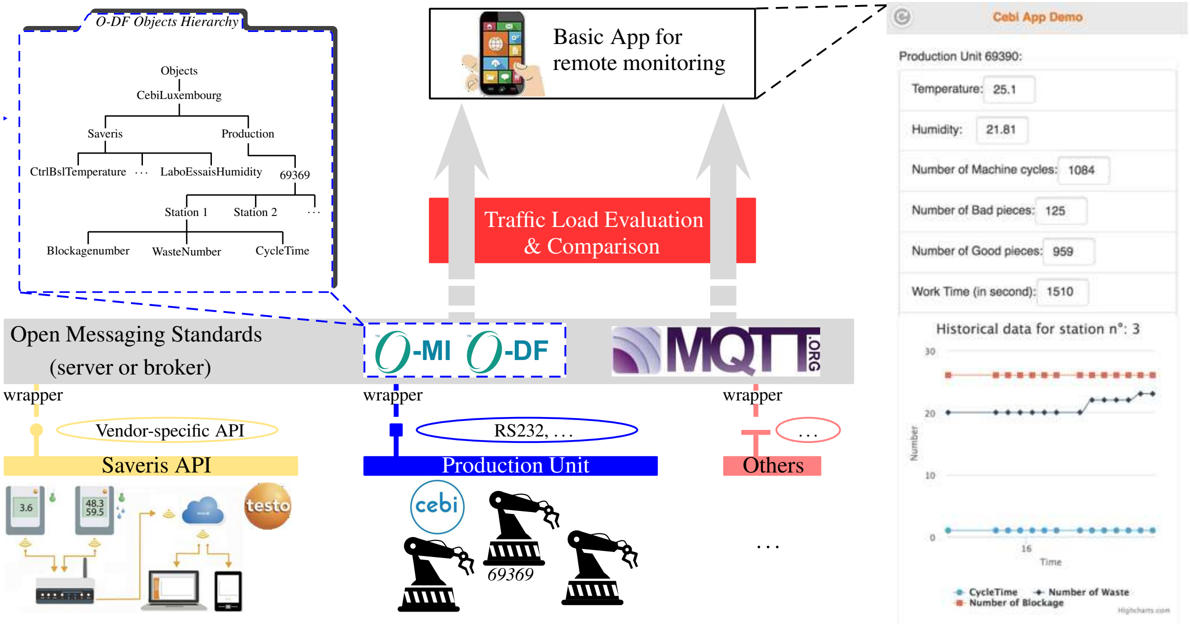
$$ER = \frac{\ell_{\text{payload}}}{TL} \tag{8}$$

C. O-MI/O-DF: a reminder of the traffic load analytical model

A specific methodology has been introduced in [6] to build the analytical model of the O-MI/O-DF standards. The same methodology is followed in this paper to build the analytical model of the MQTT standard. As explained before, O-MI relies on the HTTP protocols. The application layer size can thus be expressed as in Eq. 9.

$$\ell_{\text{app-layer}} = \ell_{\text{HTTP}} + \ell_{\text{O-MI}} + \ell_{\text{payload}} \tag{9}$$

Fig. 3, associated with TABLE I, remind the formulas needed for computing the size of (i) the HTTP header (i.e., ℓ_{HTTP}); (ii) the O-MI requests/responses (i.e. $\ell_{\text{O-MI}}$), (iii) the O-DF payload (i.e. ℓ_{payload}). Let us note that the traffic load is



Production Unit 69390:
Temperature: 25.1
Humidity: 21.81
Number of Machine cycles: 1084
Number of Bad pieces: 125
Number of Good pieces: 959
Work Time (in second): 1510

Fig. 4. Industrial Use Case Scenario

computed thanks to the expression $TL = L_{req} + L_{rep} + L_{ack}$ (since O-MI/O-DF will use only one message for the request and one for the response from an application perspective), where L_{ack} is defined as follows: $L_{\text{ack}} = \left(\left\lceil \frac{n_{\text{req}}}{m} \right\rceil + \left\lceil \frac{n_{\text{resp}}}{m} \right\rceil\right) \cdot \ell_{\text{low-layer}}$, since there is no application acknowledgments.

III. INDUSTRIAL CASE STUDY: O-MI/O-DF VS. MQTT

The overall use case is depicted in Fig. 4, which involves a company expecting to publish industrial plant-related information through an O-MI server or a MQTT broker in order to create a basic application for monitoring the production environment. Let us assume that only two vertically-oriented closed and proprietary systems exist: one for the plant metrology (based on the technology Saveris) and the second one for the production itself (information can be accessed directly from each production unit through a Programmable Logic Controller – PLC). An agent (or wrapper) has been developed and implemented on the O-MI server and the MQTT broker for translating the information coming from these two systems and making them compliant with the O-MI or MQTT standards. The performance evaluation and comparison take place between the O-MI server/MQTT broker and the application. As the application is intended to be used for monitoring these two systems, all the data needs to be collected on this application.

Even though the software agents between the proprietary systems and the server/broker are not presented in this paper, it is necessary to define the data structure that needs to be communicated in the application. Fig. 4 shows also the generic O-DF Objects hierarchy built for this scenario. This hierarchy highlights that `Saveris` and `Production` are defined as O-DF 'Object', inside the main Object `CebiLuxembourg` (corresponding to the name of the company). In the `Saveris` Object, the Object properties (called InfoItems) have been defined so as to correspond to the sensor information. In the `Production` Object, different O-DF 'Object' are nested; for instance, `69369` corresponds to the number assigned to the considered production unit, which is divided into many stations. And finally, InfoItems named '`Blockagenumber`', '`WasteNumber`', . . . , provide information about the status of each station. To be consistent with this structure as well as for comparison purposes, the MQTT topics are based on the O-DF paths defined by this hierarchical structure, as MQTT does not specify, impose, nor recommend any data structure. For example, the topics are: '`Objects/CebiLuxembourg/Saveris/CtrlBslTemperature`', . . . , '`Objects/CebiLuxembourg/Saveris/LaboEssaisHumidity`','`Objects/CebiLuxembourg/Production/Station 1/Blockagenumber`, and so on. Overall, it corresponds to 21 Objects and 78 InfoItems in the O-DF structure, and to 78 topics in MQTT, which makes it easy to compare both standards. The first analysis is based on the industrial setting above-introduced, and the second one evaluates the impact of using aggregation-like mechanisms.

A. *Industrial setting analysis*

In this scenario, we assume that the application – for plant monitoring – periodically requests all the data hierarchy/topics. The application sends O-MI `Immediate Read` requests (one of the operations defined in Fig. 3) to the O-MI server, by specifying either the whole O-DF structure or only part of it as payload. In order to minimise the request size, the O-DF root 'Object' is only embedded as payload in the O-DF read request. Following this request, response messages containing the "Values" of the requested InfoItems are pushed to the

TABLE II
OVERALL TRAFFIC LOAD (IN BYTES) ON THE INDUSTRIAL SETTING: O-MI/O-DF VS. MQTT

Layers	**O-MI/O-DF**	**MQTT**
Payload (value)	503	503
Data presentation (i.e. O-DF structure or MQTT Topics)	5341	2954
Messaging protocol (i.e. O-MI or fixed part & length of MQTT)	315	160
HTTP - *if needed*	48	0
TCP	240	3200
IP (v4)	240	3200
Network access methods (Ethernet)	312	4160
Overall Traffic Load (*Efficiency Ratio*):	**6999 (*83.5%*)**	**14177 (*25.5%*)**

application. On the other hand, the application subscribes to all MQTT topics by sending a `Subscribe` request by containing the following aggregated topic `Objects/xx...xx/#` in order to minimise this request as well (otherwise, as many requests as topics should be sent). Following this request, the MQTT broker pushes a `Publish` response containing the "Values" of each topic (each time it receives a notification from the system).

Based on this scenario, the analytical model of the traffic load detailed in the section II are applied and the associated results are shown in the TABLE II. The following conclusions can be drawn:

- The overall traffic load is much less important (around twice less) when using O-MI/O-DF rather than MQTT. Indeed, the O-DF structure (especially in the response) is based on an aggregation-like mechanism that plays a major role in minimising the overall traffic load (compared with MQTT, which does not implement such kind of mechanism). Even though the overall length of the MQTT data presentation (considered only as the topics) is also much less important than in O-MI/O-DF, the transport of these data needs to use as many TCP segments as topics (i.e., 80 by counting the subscription request and the application acknowledgement of this subscription without taking into account the TCP acknowledgements) instead of using only 6 TCP segments when using O-MI/O-DF.
- From an efficiency ratio perspective, it can be noted that this ratio is more than three times more important for O-MI/O-DF than for MQTT. Let us note that the data presentation part (in addition of the values themselves) is also considered as payload for computing this ratio. Indeed, we consider that the generic data model (with the use of semantic vocabularies) introduced in O-DF is beneficial for representing IoT data/service in order to integrate more complex reasoning out of them. Even though MQTT can give some information about the hierarchy (through the topic structuration), MQTT does not allow to add metadata to the sensors' values (e.g, unit of the value, accuracy of the sensor, and so on).

B. Aggregation- vs. non aggregation-like mechanism

In this section, the objective is to set up the size of the data structure – *O-DF parameters such as the Object's name, InfoItems's name and values, as well as MQTT parameters such as the topics' name and values* – in order to assess the impact of the number of O-DF InfoItems or MQTT topics on the Traffic Load, and accordingly the impact of using an aggregation-like mechanism. To do so, it is important to increase only one parameter (i.e. the number of InfoItems/topics) at the same time. Based on the lengths defined in the real-life scenario, the average length for each parameters is considered, namely $\ell_{objID} = 8\ bytes$, $\ell_{name} = 14\ bytes$ and $\ell_{value} = 6\ bytes$ (i.e., that all objects name/InfoItems name/values have respectively the same length. As the request does not change with the previous scenario (requesting all available information), it means that the first response only contains one InfoItem for O-MI/O-DF and only one topic for MQTT, the second only two (first) InfoItems/topics, and so on, until reaching the whole InfoItem hierarchy/topics (78 in total).

Fig. 5 gives insight into the Traffic Load evolution, along with the number of frames sent according to the number of O-DF InfoItems or MQTT topics. It can thus be noted that:

- If the number of InfoItems/topics to be collected is inferior to 10, then the aggregation mechanism of O-MI/O-DF is not efficient in terms of traffic load since the traffic load in MQTT is less important in this case.
- On the contrary, if the number of InfoItems/topics to be collected is superior to 10, then the aggregation mechanism of O-MI/O-DF is more efficient.

However, this conclusion should also be put into perspective with the fact that, the higher the number of MQTT topics, the higher the number of messages. But at the same time, the higher the number of O-DF InfoItems, the bigger the size of the O-MI/O-DF response. This is also important to be noted, as the traffic load will significantly increase in case of frame error occurrences (since TCP needs to retransmit the frames in error). All these considerations must be taken into account when designing a network, including more complex functionalities such as the network control (e.g., for adapting it to the demand).

IV. TOWARDS PERFORMANCE-DRIVEN NETWORK DESIGNS IN INDUSTRIAL CYBER-PHYSICAL SYSTEMS

Industrial CPS is gaining a growing attention in both the academic and indutrial sectors. With an increasingly trend to digitalize all aspects of companies, it is an important shift of paradigm that is leading to a new way to design the industry. Lower level systems (in the automation pyramid) needs to be visible at the higher level, but also accessible from the outside world for creating more flexible, agile and smarter services for the industry. It consists of developing digital shadow(s) of the vertically-oriented closed and proprietary systems by relying to the best extent possible on open and standardised technologies, standards and frameworks. All these shadows form a complex and interlinked System, also referred to

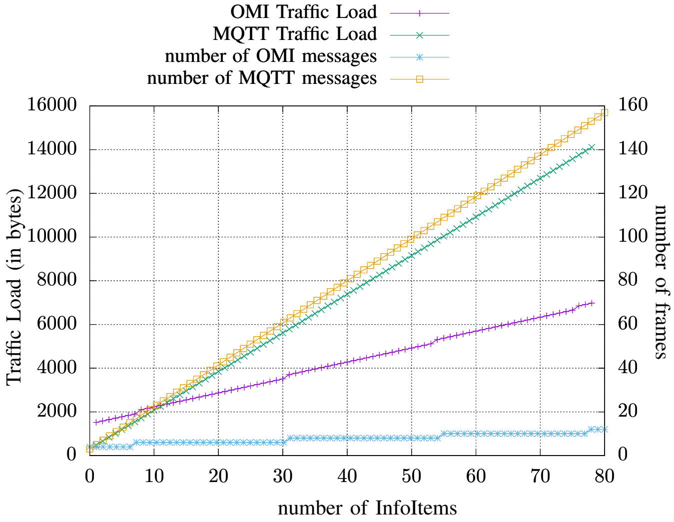


Fig. 5. Traffic Load evolution vs. Number of O-DF InfoItems

as "System-of-Systems" in the scientific community. Those systems need to communicate and cooperate, in real time, with each other and potential human beings. To do so, it is of the utmost importance to have a deep understanding to what extent these new protocols perform, and to what extent they impact on the network performance (e.g., in terms of traffic load). This is important, as it helps system designers to properly (re-)think and adapt the design of the network by assessing the compliance with the QoS requirements (freshness, delay, ...), while supporting more dynamic structures. Dynamic refers here to the fact that the network has the ability to adapt itself to new production demands or fault recovery purposes. For example, if the production planning is frequently modified due to the use of the "*order your personalised product online to get it tomorrow*" paradigm, the network needs to be online reconfigured. For the design of the network, it means that the control functions, which are currently running on field controllers, become distributed on software components rather than on dedicated hardware devices. Such capabilities are offered by paradigms such as SDN (Software-Defined Network) and NFV (Network Functions Virtualization).

In addition to succinctly presenting two open and standardised protocols for Machine-to-Machine communications (MQTT and O-MI/O-DF) through their assessment in a real-life industrial case-study, this paper provides some conclusions and trends regarding the use of aggregation-like mechanisms. Those trends could be used, as first rules, to control what mechanisms (or protocols) should be used for transporting real-time data according to current status of the network (current traffic load, noisy environment, ...) and potential demands (in terms of number of requested data items/topics). Those rules could be implemented in an IoT gateway, as a PONTE-like bridge (https://www.eclipse.org/ponte/), which handles interoperability issues by selecting the appropriate protocol and associated request/response messages.

V. CONCLUSION

Increasing connectivity between the technologies, systems and processes is the cornerstone of their abilities to communicate and cooperate, in real time, with the human beings in the loop. It mainly consists of developing digital shadow(s) of the vertically-oriented closed and proprietary systems by relying to the best extent possible on open and standardised technologies, standards and frameworks. However, those digital shadows use new communication protocols (such as O-MI/O-DF or MQTT, both presented in this paper), which their performance and their impact on the network (e.g., in terms of traffic load) have to be analysed for designing properly the IT infrastructure.

Conclusions and trends highlighted in this paper, in particular regarding the use of aggregation-like mechanism, open up the opportunity to implement new software controllers (instead of hardware components). The network will therefore has the ability to adapt itself to new production demands (or fault recovery). Such capabilities could rely on appealing paradigms such as SDN (Software-Defined Network) and NFV (Network Functions Virtualization).

REFERENCES

[1] W. A. Khan, L. Wisniewski, D. Lang, and J. Jasperneite, "Analysis of the requirements for offering industrie 4.0 applications as a cloud service," in *Industrial Electronics (ISIE), 2017 IEEE 26th International Symposium on*. IEEE, 2017, pp. 1181–1188.
[2] J. Schlechtendahl, M. Keinert, F. Kretschmer, A. Lechler, and A. Verl, "Making existing production systems industry 4.0-ready," *Production Engineering*, vol. 9, no. 1, pp. 143–148, 2015.
[3] T. Sauter, "The three generations of field-level networks - evolution and compatibility issues," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, November 2010.
[4] S. Kubler, J. Robert, A. Hefnawy, C. Cherifi, A. Bouras, and K. Främling, "Iot-based smart parking system for sporting event management," in *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. ACM New York, 2016, pp. 104–114.
[5] C. D. Turnitsa, "Applying the levels of conceptual interoperability model in support of integratability, interoperability, and composability for system-of-systems engineering." *Journal of Systemics, Cybernetics and Informatics*, 2007.
[6] J. Robert, S. Kubler, Y. Le Traon, and K. Främling, "O-mi/o-df standards as interoperability enablers for industrial internet: A performance analysis," in *Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE*. IEEE, 2016, pp. 4908–4915.
[7] L. Năstase, I. E. Sandu, and N. Popescu, "An experimental evaluation of application layer protocols for the internet of things," *Studies in Informatics and Control*, vol. 26, no. 4, pp. 403–412, 2017.
[8] L. Durkop, B. Czybik, and J. Jasperneite, "Performance evaluation of m2m protocols over cellular networks in a lab environment," in *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*. IEEE, 2015, pp. 70–75.
[9] T. O. Group, "Open-Messaging Interface (O-MI) specification: https://publications.opengroup.org/c14b," 2017.
[10] ——, "Open-Data Format (O-DF) specification: https://publications.opengroup.org/c14a," 2017.
[11] K. Främling, S. Kubler, and A. Buda, "Universal messaging standards for the iot from a lifecycle management perspective," *IEEE Internet of things journal*, vol. 1, no. 4, pp. 319–327, 2014.
[12] J. Robert, S. Kubler, N. Kolbe, A. Cerioni, E. Gastaud, and K. Främling, "Open IoT ecosystem for enhanced interoperability in smart citiesexample of métropole de lyon," *Sensors*, vol. 17, no. 12, p. 2849, 2017.
[13] OASIS Standard, "Mqtt version 3.1. 1," http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf, 2014.